

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



REPORT: Operating System

Lab 5

Thread

Lecturer: Mr. Vũ Văn Thống

Members: Phạm Nguyễn Thái Khương - 1812694

TP. HỒ CHÍ MINH, THÁNG 05/2020

1. What resources are used when a thread is created? How do they differ from those?

Khi một luồng được tạo ra thì luồng không cần thêm bất cứ tài nguyên nào để thực thi vì các luồng chia sẻ tài nguyên như bộ nhớ của process mà nó thuộc về. Lợi ích của việc chia sẻ đó là cho phép một ứng dụng có vài luồng hoạt động khác nhau nhưng cùng dùng một vùng không gian bộ nhớ. Trong khi đó, việc tạo process rất tốn kém vì nó luôn cần một vùng không gian bộ nhớ khác được cấp phát. Và cho dù các process đó có chia sẻ bộ nhớ thì việc giao tiếp cũng rất khó khăn so với sử dụng thread.

2. Is it possible to have concurrency but not parallelism? Explain.

Có thể có concurrency mà không parallelism.

Concurrency (Tương tranh) là hai việc hay luồng khác nhau bắt đầu làm việc trong một khoảng thời gian chồng chéo nhau, tuy nhiên, nó không có nghĩa là cần phải chạy cùng lúc.

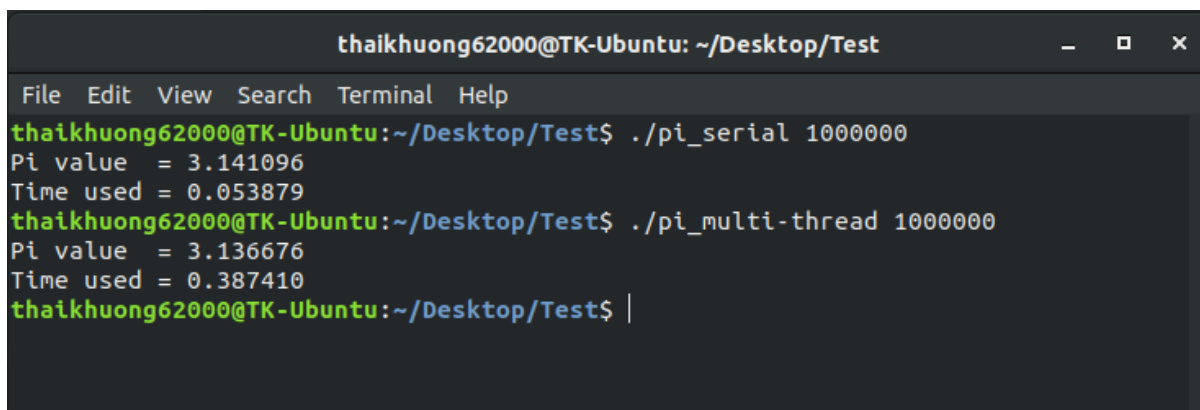
Ví dụ: Hai process A,B cần làm các việc khác nhau P1, P2, P3. Thì khi P1 của A chạy, sau đó P1 của B chạy. Sau đó P2 của A chạy... được gọi là concurrency.

Parallelism (Song song) là hai hoặc nhiều việc bắt đầu thực thi cùng lúc (đồng thời).

Ví dụ: Hai process A,B cần làm các việc khác nhau P1, P2, P3. Khi P1 của A và P1 của B cùng thực thi đồng thời được gọi là parallelism.

Trong ví dụ trên, sẽ có lúc các process chuyển đổi nhanh chóng. Và khi đó xảy ra tương tranh nhưng không đồng thời.

3. Problem 1:



```
thaihuong62000@TK-Ubuntu: ~/Desktop/Test
File Edit View Search Terminal Help
thaihuong62000@TK-Ubuntu:~/Desktop/Test$ ./pi_serial 1000000
Pi value = 3.141096
Time used = 0.053879
thaihuong62000@TK-Ubuntu:~/Desktop/Test$ ./pi_multi-thread 1000000
Pi value = 3.136676
Time used = 0.387410
thaihuong62000@TK-Ubuntu:~/Desktop/Test$ |
```

Threaded code cần phải thực hiện cùng lượng việc như serial code nhưng nó có thêm gánh nặng từ việc điều phối luồng và chuyển đổi ngữ cảnh.

Chương trình đa luồng có thể sẽ nhanh hơn so với đơn luồng nếu như có đủ CPU để chia sẻ công việc.